REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden to Washington Headquarters Service, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington, DC 20503.
PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.

1. REPORT DATE *(DD-MM-YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From - To)*
AUG 2011	**Conference Paper (Post Print)**	APR 2009 – JAN 2010

4. TITLE AND SUBTITLE

EMERGING NEUROMORPHIC COMPUTING ARCHITECTURES AND ENABLING HARDWARE FOR COGNITIVE INFORMATION PROCESSING APPLICATIONS

5a. CONTRACT NUMBER
IN-HOUSE

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

Robinson E. Pino (AFRL), Gerard Genello (AFRL), Morgan Bishop (AFRL), Michael J. Moore (ITT), Richard Linderman (AFRL)

5d. PROJECT NUMBER
NEUR

5e. TASK NUMBER
PR

5f. WORK UNIT NUMBER
OJ

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
ITT/AES
775 Dandelion Drive
Rome NY 13441

8. PERFORMING ORGANIZATION REPORT NUMBER
N/A

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

AFRL/RITC
525 Brooks Road
Rome NY 13441-4505

10. SPONSOR/MONITOR'S ACRONYM(S)
N/A

11. SPONSORING/MONITORING AGENCY REPORT NUMBER
AFRL-RI-RS-TP-2011-6

12. DISTRIBUTION AVAILABILITY STATEMENT
APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED. PA #: 88ABW-2010-0293
DATE CLEARED: 22 JAN, 2010

13. SUPPLEMENTARY NOTES
Publication in Cognitive Information Processing (CIP), 2010, on 14-16 Jun 2010, pp 35-39. Print ISBN: 978-1-4244-6457-9 . This work is copyrighted. One or more of the authors is a U.S. Government employee working within the scope of their Government job; therefore, the U.S. Government is joint owner of the work and has the right to copy, distribute, and use the work. All other rights are reserved by the copyright owner.

14. ABSTRACT
The highly cross-disciplinary emerging field of neuromorphic computing architectures for cognitive information processing applications requires knowledge within many research fields: computer architecture, neuroscience, cognitive psychology, cognitive modeling, dynamical systems, belief systems, software, computer engineering, etc. In our effort to develop cognitive systems atop a neuromorphic computing architecture, we explored the issues associated with mapping computing strategies such as the Brain State-in-a-Box and Confabulation within a Cell-BE powered 54 TeraFlops high performance computer Linux cluster. In this work, we seek to understand the underlying mechanisms for emulating neuromorphic-based cognitive process and their computational scaling properties towards human-like cognition and perception.

15. SUBJECT TERMS
Neuromorphic, Cognitive, Computing, Emerging Technology, Computational Intelligence

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			ROBINSON E. PINO
U	U	U	UU	6	19b. TELEPHONE NUMBER (*Include area code*) N/A

Emerging Neuromorphic Computing Architectures & Enabling Hardware for Cognitive Information Processing Applications

Robinson E. Pino [#1], *Senior Member, IEEE*, Gerard Genello [#2], *Fellow, IEEE*, Morgan Bishop [#3], Michael J. Moore [*1], and Richard Linderman [#4], *Fellow, IEEE*

[#] *Advanced Computing Architectures, United States Air Force Research Laboratory*
525 Brooks Road, Rome, NY 13441 USA
[1] robinson.pino@rl.af.mil
[2] Gerard.Genello@rl.af.mil
[3] morgan.bishop@rl.af.mil
[4] richard.linderman@rl.af.mil

[*] *ITT/AES*
775 Dandelion Drive, Rome NY 13441 USA
[1] mike.moore@itt.com

***Abstract*—The highly cross-disciplinary emerging field of neuromorphic computing architectures for cognitive information processing applications requires knowledge within many research fields: computer architecture, neuroscience, cognitive psychology, cognitive modeling, dynamical systems, belief systems, software, computer engineering, etc. In our effort to develop cognitive systems atop a neuromorphic computing architecture, we explored the issues associated with mapping computing strategies such as the Brain State-in-a-Box and Confabulation within a Cell-BE powered 54 TeraFlops high performance computer Linux cluster. In this work, we seek to understand the underlying mechanisms for emulating neuromorphic-based cognitive process and their computational scaling properties towards human-like cognition and perception.**

I. Introduction

The goal of engineering information systems with cognitive human-like skills has challenged scientist and engineers for many years. Brain anatomical networks are sparse, complex, and have economical small-world properties [1]. In other words, brain networks have characteristically small-world properties of dense or clustered local connectivity with relatively few long-range connections mediating a short path length between any pair of neurons or regions in the network [1]-[4]. Neurons and their synapses are well accepted as the basic functional components of a brain, just as switches are the basic component of a digital computer. Neurons have inputs, called dendrites, and outputs called axons. In many regions of the cortex, neuronal response properties remain relatively constant as one moves perpendicular to the surface of the cortex, while they vary in a direction parallel to the cortex [5]. Such columnar organization is particularly evident in the visual system, in the form of ocular dominance and orientation columns [5]. According to the hypothesis [6], the cortex is modularized into regions of varying size wherein the internal structure, function and external connectivity are similar, and that these regions of similarity are assemblies of cortical columns which represent another level of modularity because of how they connect. In this way a cortical column may be an architectural building block useful for understanding cognitive functions. Connectivity complexity reduction is supportive of the argument that cortical column hypothesis provides good middle ground for exploring brain computational architecture. However, the computational architecture is expected to be diverse. Our research efforts focuses on two main objectives: First, investigate alternative cortical column models. Specifically, investigate models proposed by Hawkins (a Bayesian model [7]), and Anderson (a network of point attractors [8]). Second, evaluate the performance of large scale cortex models by simulation and emulation within a Cell-BE powered 54 TeraFlops high performance computer Linux cluster.

II. Experimental Details

A. High Performance Computing

For neuromorphic modeling and emulation, we presently have at our disposal a 288 node Cell-BE cluster organized as 12 subnets, each with 24 nodes. Each subnet has a dual quad 3GHz Xeon processor head node. Each node has six available SPE vector processors and a dual core 78 MHz PPE (Power PC). There are 1728 SPEs in total. Each SPE is capable of slightly more than 25 GFLOPS for a total Cell-BE cluster capability of 43.2 TFLOPS, not accounting for head node and PPE contributions. GNU C++ development tools were used to develop the emulator, and a Publication/Subscription message passing system was used for communication within the emulation. Network interconnectivity is 10 Gigabit Ethernet. This 2400 core (Xeons + Power PCs + SPEs) facility's processors are somewhat specialized. The head nodes are conventional general purpose platforms with 32 GB of memory (each). The CELL-BE PPEs, also general purpose, each have 228 megabytes of RAM. The SPEs are specialized

to be vector processors; they each have about 128Kbytes of useable RAM. Very fast DMA channels within a CELL-BE move data between main memory (PPE) and SPE memory. The Xeons, and PPEs, run Linux; the SPEs are essentially naked (no operating system), but interact with each other and the PPEs using DMA channels, interrupts and semaphores.

B. Model Description

The Brain State-in-a-Box (BSB) algorithm was selected as the attractor function to incorporate into the network study because of its association with the Ersatz Brain project [8]. Ersatz Brain is an effort to model aspects of mind with nested networks of fixed point attractors. BSB uses state vectors with "N" real numbers in the range of (-1.0...+1.0). Its name is a metaphor for describing the algorithm as an N dimensional shape. Its fixed basin points of attraction lie in its corners. An N dimensional BSB function can separate M basin points, where M is ~15% of N. The model has many applications including machine reading, author ID, and scene interpretation. Applying the model efficiently involves exploring architecture design space, implementations, and evaluations of neuromorphic computing models. Preliminary assessment of the attractors suggested these attractors were useful for recognizing features using feed forward (afferent) data as well as feedback (expectation) data.

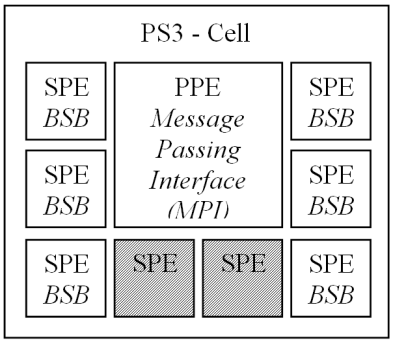

Fig. 1. Task distribution of one Cell-BE node.

TABLE I
PERFORMANCE, POWER, COMMUNICATION AND MODELING CAPABILITIES: 1 VS. 288 CELL-BE NODES

Number of Cell-BE Nodes	1	288
Peak computational performance achievable by BSB application	102 GFLOPS	29.4 TFLOPS
Number of 128-dimensional BSB models supported (10ms reaction time)	3,000	864,000
Equivalent mini-columns in the visual cortex of the human brain	140 W	3,456,000*
Total power consumption	140 W	40KW
Achieved total network bandwidth for the communication test using MPI	~1Gb/s	~12Gb/s

*The V1 layer of the visual cortex consists of about 1,600,000 minicolumns.

Details of implementing a 128-dimensional BSB model on the Cell processor can be found in [9][10]. Referring to Figure 1, in the large-scale BSB model implementation, 128-dimentional BSB models are run on each of the six Synergistic Processing Elements (SPEs) on the Cell processor. The data communication functions are implemented on the PowerPC Processing Element (PPE), and the word and sentence level confabulation models are implemented on cluster head nodes associated with groups of 24 Cell-BE nodes. The BSB model was also implemented in an FPGA hardware version that achieved ~150 speedup over software [10][11]. Table I shows a comparison of the computing performance, communication performance, power consumption, and modeling capabilities between a single Cell-BE with 6 SPEs and the entire cluster (288 nodes). Theoretically, we can implement two V1 layers of the human visual cortex on this cluster.

C. Confabulation Model

An investigation of confabulation surfaced reports by Robert Hecht-Nielson of a cognitive mechanism which explains all of cognition [12]. The center piece of his reports both published and in presentations, was a demonstration of software which completed sentences with no context, and another which completed a sentence in the context of two other sentences. The hypothesis is that the reported algorithm models the fundamental cognitive mechanism, and that the mechanism must be somehow layered on a large scale (many interconnected confabulators) to produce a level of coherence. The algorithm is computationally similar to Bayesian Belief, but it does not use a Belief tree network. It was decided to explore Confabulation first in its reported context (textual data) and to consider it later on as a candidate for extra striate (above V1) modelling, fulfilling an expectation role.

The reported sentence completion algorithm trained by reading text; lots of text. It then "recalled" by using a context (for example, the start of a sentence) to retrieve a sequence of words and phrases which its training statistically connected to the context. The training consisted of reading one sentence at a time and breaking it into sequences of words and phrases - all possible combinations of these. Sentence by sentence the training keeps track of all words and phrases encountered, and all sequences formed, through statistical links.

III. RESULTS AND DISCUSSION

The use of IBM Cell-BE technology (Sony PlayStation® 3 platform or PS3 for short) to accelerate BSB performance was investigated. Runtime measurements show that we have been able to achieve about 70% of the theoretical peak performance of the processor when implementing a 128 element vector using a matrix shuffle strategy to improve Cell-BE SPE instruction utilization [9].

The 128 element BSB recall algorithm was implemented on a single SPE element of the Cell-BE architecture. The complexity is 33,280 FLOPs/ recursive cycle. Ten cycles are needed for convergence yielding 332,800 FLOPs/ recall. Peak efficiency corresponds to all floating operations being

performed as quad word operations, with all other (non-floating point) instructions executing in the parallel instruction pipe. In this case, peak is 332,800/4 = 83,200 Quad Floating ticks. Each recall needs a weight vector load, a state vector load and a state vector unload (66,560 bytes) equivalent to 4160 quad word transfers (one quad word per tick). Compute to DMA peak ratio is therefore 83200/4160 = 20. Double buffering was used to overlap data transfer of weight matrices and state vectors with processing. Six BSBs can be run in parallel on a Cell-BE version of the platform. Efficient implementation on an SPE requires careful attention to aligning data for maximum effectiveness of intrinsic functions. Loop unrolling is essential as well to maintain the dual pipeline SIMD efficiency.

The 32 element BSB recall algorithm performs about 2240 floating operations for each recursive cycle; 2,176 for the actual algorithm and 64 for state vector conversions from and to integer fixed point. About 5 cycles are needed for convergence, yielding 11200 operations per 128 bytes of DMA data movement (no weight vector movement, and the state vector is actually 2 byte fixed point). Peak FLOP rate is (2176/4 + 64) 608 Quad Floating ticks/cycle. The peak DMA rate is (128/16) 8 DMA ticks. The peak compute to DMA ratio is therefore 608/8 = 76.

We researched, implemented, and evaluated the performance of the confabulation model, focusing specifically for two example application problems that we call here sentence completion and intelligent on-line character recognition (OCR). In both of these applications the basic problem is to complete a partial natural language sentence in a plausible, sensible way, given that only a fragment of the input sentence is available, and given that the system has been trained by exposure to a large training corpus of textual electronic media (e.g. books and news feeds). Good solutions to the sentence completion problem could very well translate to other input modalities (i.e. audio and imagery), and map to solutions in several higher level application scenarios.

Three strategies were explored for optimization of the sentence completion algorithm: software optimization, software analysis and hardware architecture augmentation. Our analysis shows there is potential to improve the three structure techniques using hashing strategies. The hashing strategies may improve data locality as well. A hash version of training was demonstrated in about 4 seconds, compared to the 45 seconds the tree structures used. The cogent confabulation algorithm is an ideal candidate for parallel processing. It also shows that although increasing the number of processors or the size of memories can increase the performance of training and recall, the relations between resource cost and performance associated with these variations are not always linear. The details of hardware configuration must be carefully considered to achieve good cost performance tradeoffs. We suggest that this work can be extended to more complex implementations of confabulation systems.

We performed experiments to test the confabulation recall algorithm by using it to complete a number of short arbitrary 'starter' sentences. The results are show in Table II, were the "best" completed sentences, which typically appeared on the first or second attempt (successive completions using the same starter sentence can be different because the algorithm has a stochastic element). The results were typically not text strings appearing in the training text. However, sometimes they were exactly the same as sentences in the training text (e.g.: "That it is …" and "Even the …").

TABLE II
SELECTED SENTENCE CONFABULATIONS

Initial Sentence	Output Sentence
I am	I am dear Sir Your most humble servant and state and I give not only health and salvation to the
Come now	Come now what was this duel about ?
Finally the	Finally the amendment was negative
For this purpose	For this purpose he said that the object of child and difficulties which difficulties were increased by the
Think of the people	Think of the people and ask yourself whether the world is more likely to be a believer in the unity

In general sentences completed by the confabulator were similar in style to those in the training material. This effect was magnified by training networks on material from only one style or author. For example, when trained only on material from Shakespeare, using a short, general starter sentence produced the following completion that was not in the training test, but reflects the style of Shakespeare.

Original: "Go to the forge with it then shape it I would not have things cool."

Starter: "Go to"

Completion: "Go to me at your convenient leisure and you shall know how I speed and the conclusion shall be "

Similar effects were seen with training limited to JFK's speech, the Dr. Seuss material, and religious material. This model uses totally unsupervised training methods to store probabilistic representations of token sequential patterns in sentences, with no semantic extraction or analysis methods, yet its ability to complete sentences based on partial data with nearly correct grammar is arguably a 'cognitive like' feature.

One idea developed as a result of the investigation of the confabulation model was to regard it as a means to model prediction effects of "higher neocortex." For example, confabulation might be useful for modeling V2 while improving the V1 Model; V1 needs V2 because V2 provides predictive influences on perception. Likewise confabulation might later model medial-temporal lobes and later yet, frontal. An experiment was designed to investigate how that might be implemented. In this experiment, BSBs were used to recognize individual characters in text strings. To make this challenging, many characters were deleted or smudged out. The confabulator was used as a mechanism to provide guidance to the BSBs in order to fill in missing information. 256 element BSBs were trained to recognize multiple character fonts using a 16X16 pixel array. The results of the

"BSB initial pass" interpreting the text are passed on to a two layer confabulator: a word level and a phrase level. Characters which could not be recognized by the BSB in a limited number of iterations were passed to the word level as "unknown." Sentence length limited to was 20 words. The confabulation model (Figure 2) was trained using a list of 58,000 most common English words, and a set of 72 novels from classic literature, with an estimated total of 37 million words.

Recall performance using starter sentences drawn from trained sentences with 20% missing characters was almost perfect (~99% correct). Sentence recall using starter sentences not previously trained was in the range of 90% correct word selection and 60% correct sentence completion when 10% of the letters were missing; 24% and 86 % respectively when at 20% of letters were missing. Here by 'correct word selection' we mean that all unknown characters in a word were chosen correctly, and by 'correct sentence completion' we mean that all unknown words in the sentence were completed correctly. The actual system is more sophisticated than the simplified example shown in Figure 2. Besides lexicons for single letters and single words, it also has lexicons for each adjacent letter pair and adjacent word pairs in layer 3 and layer 4 respectively. The intelligent text recognition system could potentially process scanned text images at very high speed, continuously learning from what has been read (excepting cases when uncertainty is detected), and can anticipate or predict not only the missing portion of words based character context within words, but also based on word context in other parts of the sentence.

IV. Conclusion

Neuromorphic computational architecture development is a new and accelerating field with significant promise. Individual qualifications to contribute in this domain include familiarity in multiple disciplines such as: computer architecture/technology, parallel software development, dynamical systems, neuroscience, neurology, neuropsychology, and agent based expert systems.

The results suggest topographically organized cortex, like "early" vision, audition and tactile sensing, can be emulated using minicolumn models similar to the hybrid model we created, and that the emulation is computationally tractable on, for example, a small number (hundreds) of Cell Broadband Engine® (Cell-BE) class chips. "Higher" cortical regions, because of plasticity needs, may require more computationally intense models, which deal with spiking dynamics and liquid state machine effects

V. Future Work

We are in the process of procuring additional Cell-BE powered Play systems to increase the total number of nodes from 288 to 2,016. The configuration will consist of 84 sub-clusters of 24 nodes per sub-cluster. Each of the 84 head nodes will also have 2 GPGPU's; one NVIDIA Tesla C1060 and one NVIDIA Tesla C2050 for a total of 168 GPGPU's. Head node candidates are still being evaluated, but by combining computational power of all other processing components the cluster will have theoretical throughput of ~500 TFLOPS or ~.5 PFLOPS. We estimate that this system will allow for the emulation of ~80% of the neocortex.

Acknowledgment

The authors would like to thank the continued support of the Advanced Computing Architectures division of the US Air Force Research Laboratory, Information Directorate.

References

[1] Achard S, Bullmore E. Efficiency and Cost of Economical Brain Functional Networks. PLoS Computational Biology Vol. 3, No. 2, e17 doi:10.1371/journal.pcbi.0030017.

[2] Watts DJ, Strogatz SH (1998) Collective dynamics of ''small-world'' networks. Nature 393: 440–442.

[3] Bassett DS, Bullmore ET (2006) Small world brain networks. Neuroscientist 12: 512–523.

[4] Kaiser M, Hilgetag CC (2006) Nonoptimal component placement, but short processing paths, due to long-distance projections in neural systems. PLoS Comput Biol 2: e95.

[5] Geoffrey J. Goodhill and Miguel A. Carreira-Perpinan, Cortical Columns, Encyclopedia of Cognitive Science, Macmillan Publishers Ltd. (2002).

[6] Mountcastle V.B. The columnar organization of the neocortex. Brain. 1997;120:701–722

[7] Bernardo, J.M., and Smith, A.F.M. (1994) Bayesian Theory, Wiley, New York.

[8] Anderson, J.A. (1993). The BSB network. Pp. 77-103 in MH Hassoun (Ed.), Associative Neural Networks, New York, NY: Oxford University Press.

[9] Wu, Qing; Mukre, Prakash; Linderman, Richard; Renz, Tom; Burns, Daniel; Moore, Michael; Qiu, Qinru; Performance Optimization for Pattern Recognition Using Associative Neural Memory. IEEE International Conference on Multimedia and Expo, 2008. On pages: 1-4. Publication Date: June 23 2008-April 26 2008.

[10] Richard Linderman. Qing Wu, Qinru Qiu, "FPGA and Cell Processor Performance Optimization for Brain-State-in-a Box (BSB) cognitive Computing", 2007 ARCS Symposium on Multicore and New Processing Technologies, Aug 2007.

[11] Qing Wu, Qinru Qiu, Richard Linderman, Daniel Burns, Michael Moore, Dennis Fitzgerald. "Architectural Design and Complexity Analysis of Large-Scale Cortical Simulation on a Hybrid Computing Platform." IEEE Computational Intelligence for Security and defense Applications (CISDA), 2007.

[12] Hecht-Nielsen R., Mechanism of Cognition. In: Bar-Cohen, Y. [Ed.] Biomimetics: Biologically Inspired Technologies, CRC Press, Boca Raton, FL (2006).

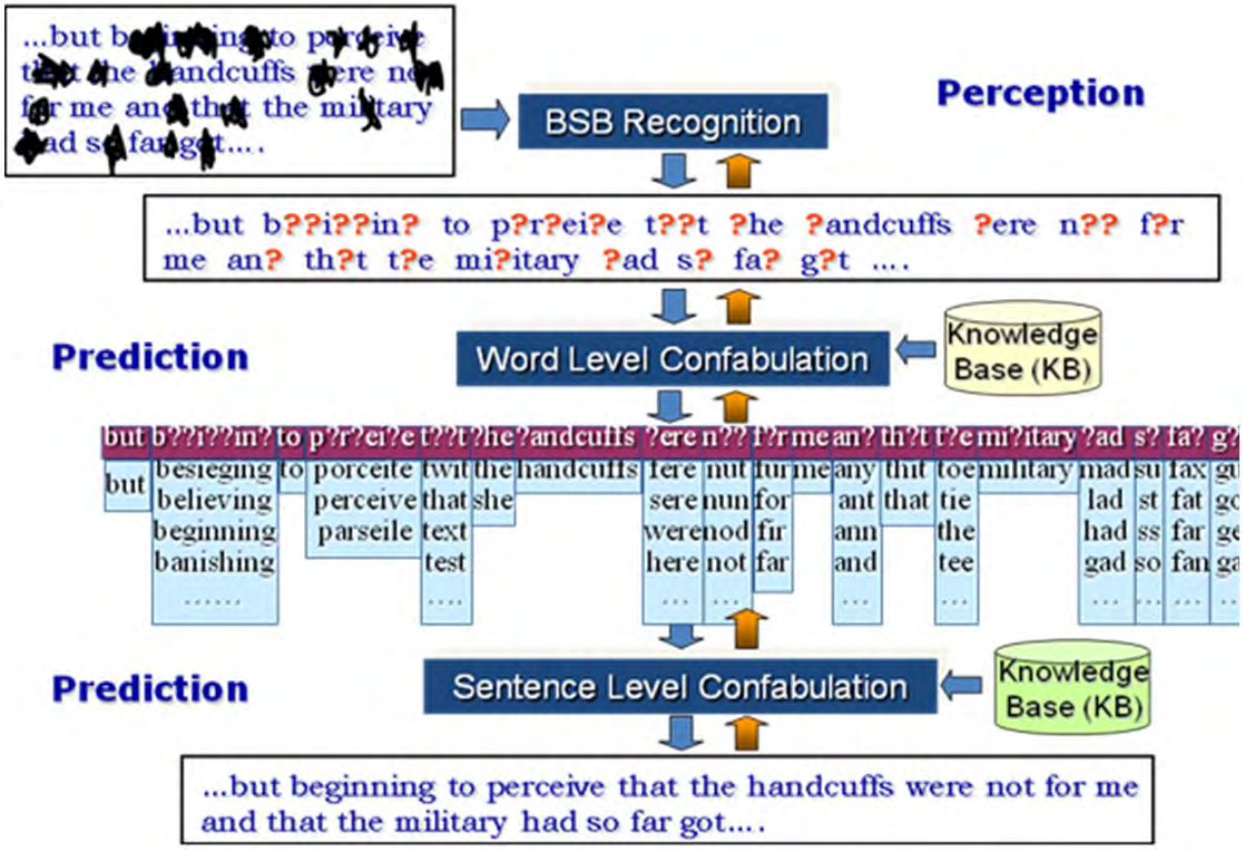

Fig. 2. The BSB/Confabulation Hybrid Model.